

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Applicant:	S. Oberst	Attorney Docket No.: MSFT120575
Application No.:	10/606,624	Art Unit: 2136 / Confirmation No.: 5087
Filed:	June 25, 2003	Examiner: D.G. Cervetti
Title:	DISTRIBUTED EXPRESSION-BASED ACCESS CONTROL	

APPLICANT'S APPEAL BRIEF

Seattle, Washington
May 27, 2008

TO THE COMMISSIONER FOR PATENTS:

This appeal brief is in support of a Notice of Appeal filed October 24, 2007, to the Board of Patent Appeals and Interferences appealing the Decision dated April 24, 2007, of the Primary Examiner finally rejecting Claims 1, 4-6, 9-13, and 16-19.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

TABLE OF CONTENTS

	<u>Page</u>
I. REAL PARTY IN INTEREST	1
II. RELATED APPEALS AND INTERFERENCES	2
III. STATUS OF CLAIMS	3
IV. STATUS OF AMENDMENTS	4
V. SUMMARY OF CLAIMED SUBJECT MATTER	5
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL.....	9
VII. ARGUMENT	10
VIII. CLAIMS APPENDIX.....	37
IX. EVIDENCE APPENDIX.....	41
X. RELATED PROCEEDINGS APPENDIX	42

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

I. REAL PARTY IN INTEREST

The subject application is owned by Microsoft Corporation of Redmond, Washington.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

II. RELATED APPEALS AND INTERFERENCES

Upon information and belief, Appellant does not have any knowledge of related appeals or interferences that may directly affect or have a bearing on the decision of the Board of Patent Appeals and Interferences (hereinafter "the Board") in the pending appeal.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206 682 8100

III. STATUS OF CLAIMS

On June 25, 2003, Appellant filed the pending patent application, including Claims 1-19. On October 11, 2006, the Examiner mailed a first Office Action rejecting Claims 1-19. On February 12, 2007, Appellant filed an amendment and response in which Claims 1, 6, and 13 were amended. On April 24, 2007, the Examiner mailed a second Office Action, finally rejecting Claims 1-19. On July 19, 2007, Appellant filed an Amendment After Final in which Claims 1, 4, 6, 9, 13, and 16 were amended, and Claims 2-3, 7-8, and 14-15 were canceled without disclaimer or prejudice. On August 1, 2007, the Examiner mailed an Advisory Action, and on October 24, 2007, Appellant filed a Notice of Appeal, which appeal follows in which Appellant entreats the Board to reverse the final rejections of Claims 1, 4-6, 9-13, and 16-19. The claims on appeal are set forth in the Claims Appendix.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206 682 8100

IV. STATUS OF AMENDMENTS

Amendments have been filed subsequent to the final rejection but were entered by the Examiner. To wit, Claims 1, 4, 6, 9, 13, and 16 were amended after the final rejections but were entered by the Examiner. Claims 2-3, 7-8, and 14-15 were canceled without disclaimer or prejudice but were also entered by the Examiner.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

V. SUMMARY OF CLAIMED SUBJECT MATTER

Regarding the claims, independent Claim 1 is directed to a networked system for accessing a piece of content. See the pending specification at p. 5, lines 13-19. The system comprises a user Web service for presenting a user having an expressed user access scope. See the pending specification at p. 5, line 15; FIGURE 3R; p. 25, lines 13-14; p. 25, line 28—p. 26, line 1; and p. 26, lines 6-9. The expressed user access scope is expressed in an accessor sentence containing dimensional extents of a security space. See the pending specification at p. 22, lines 8-9; p. 22, lines 11-13; p. 24, lines 16-19; and p. 25, lines 13-16. The system further comprises a content Web service for representing a piece of content having an expressed content access scope. See the pending specification at p. 5, lines 15-17; FIGURE 3R; p. 25, lines 22-24; and p. 25, line 28—p. 26, line 9. The expressed content access scope is expressed in a content sentence containing dimensional extents of the security space. See the pending specification at p. 22, lines 9-11; p. 23, lines 12-15; p. 24, lines 3-5; and p. 25, lines 22-25. The user Web service communicates with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope without using predicate rules, sequencing rules, and implication rules. See the pending specification at p. 5, lines 17-19; p. 12, lines 4-6; p. 12, lines 6-10; p. 15, lines 25-26; p. 16, lines 16-25; p. 16, line 27—p. 17, line 1; p. 21, lines 16-20; p. 22, lines 3-5; p. 22, lines 22-26; p. 23, lines 12-15; and p. 24, lines 14-19.

Claims 2 and 3 have been canceled without disclaimer or prejudice. Claims 4 and 5 are dependent from independent Claim 1 and are directed to further limitations of the system described above. Claim 4 is dependent on Claim 1 and recites that a compiler Web service is configured for compiling the accessor sentence and the content sentence into binary sentences, each binary sentence comprising binary phrases. Claim 5 is dependent on Claim 4 and recites an

evaluator Web service for comparing the accessor sentence and the content sentence to determine whether to grant access to the user Web service so that the user Web service can access the piece of content.

Independent Claim 6 is directed to a computer-implemented method for evaluating the scope of a content access request by a user. See the pending specification at p. 5, lines 20-27. The method comprises requesting a discovery framework by a user Web service to access a piece of content represented by a content Web service. See the pending specification at p. 25, lines 24-27; p. 29, lines 21-23; and p. 25, line 28—p. 26, line 1. The method further comprises notifying the content Web service by the discovery framework of the access request by the user Web service. See the pending specification at p. 5, lines 23-25. The method further comprises requesting the discovery framework by the content Web service for an access evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content. See the pending specification at p. 25, lines 19-21; p. 26, lines 1-9; and p. 31, lines 5-7. The access scope of the user Web service is conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service without using predicate rules, sequencing rules, and implication rules. See the pending specification at p. 12, lines 2-4; p. 16, lines 2-6; and p. 17, lines 10-16. The access scope of the user Web service is expressed by the first expression in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service is expressed in a content access scope sentence containing dimensional extents of the security space. See the pending specification at FIGURE 3F; FIGURE 3G; p. 17, lines 10-17; p. 17, lines 18-28; p. 18, lines 1-13; p. 18, lines 14-25; p. 18, line 25—p. 19, line 3; p. 20, line 20—p. 21, line 3; and p. 27, lines 1-12.

Claims 7-12 are dependent from independent Claim 6 and are directed to further limitations of the method described above. Both Claims 7 and 8 have been canceled without disclaimer or prejudice. Claim 9 is dependent on Claim 6 and recites compiling the user access scope sentence and the content access scope sentence by a sentence compiler Web service into binary sentences, each binary sentence having binary phrases, each binary phrase being a compiled dimensional extent. Claim 10 is dependent on Claim 9 and recites evaluating the binary sentences by the access evaluator service, the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence. Claim 11 is dependent on Claim 10 and recites granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero. Claim 12 is dependent on Claim 11 and recites denying access to the user Web service if one binary phrase of the resultant binary sentence is equal to zero.

Independent Claim 13 is directed to a tangible and storable computer-readable medium having computer-readable instructions that implement a method for evaluating the scope of a content access request by a user. See the pending specification at p. 5, lines 12-13. The method comprises requesting a discovery framework by a user Web service to access a piece of content represented by a content Web service. See the pending specification at p. 5, lines 22-23; p. 25, line 28—p. 26, line 1; and p. 29, lines 21-24. The access scope of the user Web service is expressed in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service being expressed in a content access scope sentence containing dimensional extents of the security space. See the pending specification at FIGURES 3F, 3G; p. 17, lines 10-17; p. 17, lines 17-28; p. 18, lines 1-13; p. 18, lines 14-25; p. 18, line 26—p. 19, line 3; p. 20, line 20—p. 21, line 3; and p. 27, lines 1-12. The method comprises notifying the content Web service by the discovery framework of the access request by

the user Web service. See the pending specification at p. 5, lines 23-25. The method further comprises requesting the discovery framework by the content Web service for an access evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content without forming an explicit relationship tying the user Web service to the content Web service via predicate rules, sequence rules, and implication rules. See the pending specification at p. 5, lines 25-27; p. 25, lines 19-21; p. 26, lines 3-9; and p. 31, lines 5-8.

Claims 14-19 are dependent from independent Claim 13 and are directed to further limitations of the computer-readable medium described above. Both Claims 14 and 15 have been canceled without disclaimer or prejudice. Claim 16 is dependent on Claim 13 and recites compiling the user access scope sentence and the content access scope sentence by a sentence compiler Web service into binary sentences, each binary sentence having binary phrases, each binary phrase being a compiled dimensional extent. Claim 17 is dependent on Claim 16 and recites evaluating the binary sentences by the access evaluator service, the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence. Claim 18 is dependent on Claim 17 and recites granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero. Claim 19 is dependent on Claim 18 and recites denying access to the user Web service if one binary phrase of the resultant binary sentence is equal to zero.

VI. GROUND OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 4-6, 9-13, and 16-19 were rejected under 35 U.S.C. § 102(e) as being anticipated in view of the teachings of U.S. Patent No. 7,080,384 (hereinafter "Wall et al.").

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

VII. ARGUMENT

As discussed below, the Examiner has failed to establish a *prima facie* case of anticipation. To establish *prima facie* anticipation of a claimed invention, each and every element arranged as in the claim, must be found in a single prior art reference. See M.P.E.P. § 2131. The applied and cited references do not teach, among many other features, the feature of:

the access scope of the user Web service being expressed by the first expression in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service being expressed in a content access scope sentence containing dimensional extents of the security space

as recited in independent Claim 6 and independent Claims 1 and 13, albeit in different manners.

The applied and cited references also do not teach the feature of:

the access scope of the user Web service being conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service without using predicate rules, sequencing rules, and implication rules

as recited in independent Claim 6. For better appreciation of the arguments below, Appellant has summarized the applied references.

A. Summary of Wall et al.

The system of Wall et al. is directed to a system of data objects and/or function objects that are linked and distributed across multiple computing devices on a computer network. Access control is provided by identifying a user of a decentralized model and assigning appropriate read, write, execute and administrative permissions to the user on a per data objects and/or function objects basis, the permissions being used to limit access to a specific subset of the data objects and/or function objects. See the Summary of The Invention of Wall et al., specifically at Col. 2, line 64, to Col. 3, line 10. See also the Abstract and Col. 3, lines 24-53 of Wall et al.

More specifically, the coupling of an identified user to an object of Wall et al. is completely different from the decoupling of users and contents of the claimed subject matter. For example, whereas the Summary of the Invention of Wall et al. explains that the identified user is assigned access on a per object basis, in the presently claimed subject matter, "the access scope of the user Web service being conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service" as recited by independent Claim 6. See also FIGURE 3R of the pending patent application. This distinguishing difference among other differences renders Wall et al. ineffective as an anticipation reference.

B. Rejection Under 35 U.S.C. § 102(e) in View of the Teachings of Wall et al.

1. Independent Claim 1

Claim 1 succinctly defines a networked system for accessing a piece of content. The system comprises a user Web service for presenting a user having an expressed user access scope. The expressed user access scope is expressed in an accessor sentence containing dimensional extents of a security space. The system further comprises a content Web service for representing a piece of content having an expressed content access scope. The expressed content access scope is expressed in a content sentence containing dimensional extents of the security space. The user Web service communicates with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope without using predicate rules, sequencing rules, and implication rules. Because the Examiner has failed to show that Wall et al. discloses every element of the claimed subject matter, no *prima facie* case of anticipation has been established.

- a. The Examiner Has Utterly Failed to Establish a *Prima Facie* Case of Anticipation by Neglecting to Show That Every Claim Limitation Is Taught by Wall et al.

Each claim limitation is neither taught nor suggested by Wall et al. As an example, the system of Wall et al. completely lacks the feature of:

a user Web service for representing a user having an expressed user access scope, the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space

as recited in independent Claim 1. The Office has indicated that the recited limitation of Claim 1 can be found in Wall et al. at Col. 5, lines 1-30. There is nothing in that location of Wall et al. that discloses the claimed feature, which Appellant recites here in full:

In a system generating an emergent model it is often desirable to represent aspects of the system using an object-oriented paradigm, in which a system is viewed as a collection of discrete objects that are self-contained collections of data structures and routines having the ability to interact with other objects in the system. Object-oriented systems provide for the definition of classes that are used to create (instantiate) objects of that class. The objects allow for the encapsulation of data and provide well defined interfaces for other objects to use when sending data to, or receiving data from, an object. One class definition can inherit the capabilities of another, allowing more complex classes to be built from simpler, underlying classes. These classes can also specify certain behaviors that are only fully defined at run-time. The combination of features provided in object-oriented systems help create a platform for hosting an embodiment of the present invention used to generate emergent models.

In the present invention, models are collections of computer instructions and data that present an interface for describing the behavior of part of a system being modeled, such that the interface is understood by other parts of the system. The present invention provides for generating an emergent model consisting of one or more models where a model consists of one or more objects, therefore a model can contain a complex hierarchy of submodels. Objects can be model inputs and outputs that are made available to relate various models to one another. These objects are

implemented using standard distributed object management techniques (e.g., CORBA, DCOM). As such, each object has a corresponding object reference.

The claimed subject matter requires an expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space. The above portions of Wall et al. seem to discuss object oriented models, which Appellant finds confusing since it has nothing to do with an accessor sentence containing dimensional extents of a security space to express the expressed user access scope.

One with ordinary skill in the art knows that there is a difference between an object and a Web service. For the sake of advancing the comprehension of the claimed subject matter by somehow assuming that there is a similarity between an object and a Web service, Appellant is unclear why such a similarity would necessarily disclose an expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space. Nothing in the above portions of Wall contains a discussion of accessor sentences, there is nothing about dimensional extents of a security space, and nothing about the use of accessor sentences to express an expressed user scope of a user Web service. If the Examiner finds at least one accessor sentence in the above cited portions of Wall et al., Appellant would be curious to know its exact location.

The Office has further indicated that the above recited limitation of Claim 1 ("a user Web service for representing a user having an expressed user access scope, the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space") can be found in Wall et al. at Col. 7, lines 15-64. There is nothing in that location of Wall et al. that discloses the claimed feature, which Appellant recites here in full:

A unit is a particular physical quantity, defined and adopted by convention, with which other particular quantities of the same kind are compared to express their value. The International System of Units (SI) defines seven base units:

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

TABLE-US-00001 length meter, mass kilogram, time second, electric current ampere, thermodynamic temperature, amount of substance mole, luminous intensity candela.

Function objects provide behavior by providing expressions that can relate data objects and function objects thereby creating networks of linked function objects and data objects. Function objects have a name, an expression, an object reference table and a solver attribute as well as access policy and trigger message propagation constraints. The expression can be thought of as a function with a plurality of inputs and output objects, which is evaluated by the solver. The expression text can be defined using various programming languages including Basic, C, C++ and Java, among others. The object reference table can contain references to data objects and function objects. The expression text can contain portions that specify method calls on the objects referenced in the object reference table. The solver evaluates the expression text, which may result in changes to objects referenced by the function object. Solvers can be implemented using compilers, interpreters or entire legacy applications. The function object solves the expression when a change message is received from one of the objects referenced by the function object. Function objects can also be solved manually by a user. Dependency constraints are optionally placed on data objects that have their values set by the expression of a function object.

An equivalence function object is an example of a function object implemented with the present invention. The following simplified example illustrates an equivalence function object that keeps several data objects equivalent (i.e., if one of the data object values changes, then the other data object values are changed such that their values match). In this example, the equivalence function is used in combination with a number data object, further providing an example of a message propagation constraint being applied.

The above portions of Wall et al. discuss "unit" as a physical quantity and function objects. Nowhere can Appellant find recited feature "a user Web service for representing a user having an expressed user access scope, the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space" of Claim 1. In Col. 8 of Wall et al., there are mentionings, in one instance, regarding a function object that has access policy constraints in which an imperative "check invoker permissions" is shown. In other words,

an invoker that invokes the function object has permissions, and those permissions are checked. Thus, the access policy constraints of the function object of Wall et al. discuss the need to check an invoker permissions but nothing about that is a revelation of an expressed user access scope, the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space" as recited by Claim 1 of the pending patent application. It is, however, a revelation of a coupling of a user to a data object or a function object in the system of Wall et al. in which permissions of a user are assigned on a per data object basis or per function object basis. Not so with the claimed subject matter of Appellant's pending patent application.

As another example of claim limitation neither being taught nor suggested by Wall et al., the system of Wall et al. completely lacks the feature of:

a content Web service for representing a piece of content having an expressed content access scope, the expressed content access scope being expressed in a content sentence containing dimensional extents of the security space

as recited in independent Claim 1. The Office has indicated that the recited limitation of Claim 1 can be found in Wall et al. at Col. 5, lines 31-64. There is also nothing in that location of Wall et al. that discloses the claimed feature, which Appellant recites here in full:

CORBA stands for Common Object Request Broker Architecture. A specification developed by the Object Management Group in 1992 in which pieces of programs (objects) communicate with other objects in other programs, even if the two programs are written in different programming languages and are running on different computing platforms. A program makes its request for objects through an object request broker (ORB), and thus does not need to know the structure of the program from where the object comes. DCOM is an acronym for Distributed Component Object Model, the version of Microsoft's Component Object Model (COM) specification that stipulates how components communicate over Windows-based networks. DCOM permits the distribution of different components for a single application across two or more networked computers, running an application distributed across a network so that the

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206 682 8100

distribution of components is not apparent to the user, and remotely displaying an application.

In the present invention, the object reference can consist of a relative Uniform Resource Identifier (URL) component and/or a unique Uniform Resource Identifier (URI) component. Distributed object management techniques provide for the resolution of an object reference into a proxy object suitable for making method invocations on the corresponding remote object. Techniques for sending messages between objects in a distributed object system are common. The present invention sends and receives change messages that contain the object reference of the source of the message as well as information indicating how the source object changed. Common to message sending techniques is the option of having the sending object block or not block while the receiving object processes the message. In the present invention, blocking message processing is used.

The claimed subject matter requires an expressed content access scope being expressed in a content sentence containing dimensional extents of the security space. The above portions of Wall et al. seem to discuss CORBA and sending messages between objects, which further confuse Appellant since it has nothing to do with a content sentence containing dimensional extents of a security space to express an expressed content access scope instead of a user access scope.

The Office has additionally indicated that the above recited limitation of Claim 1 ("a content Web service for representing a piece of content having an expressed content access scope, the expressed content access scope being expressed in a content sentence containing dimensional extents of the security space") can also be found in Wall et al. at Col. 7, lines 15-64. There is nothing at that location of Wall et al. that discloses the claimed feature, which Appellant has already discussed above.

- b. Instead of Giving the Claimed Subject Matter the Broadest Reasonable Interpretation Consistent With Appellant's Specification, the Examiner Gave Wall et al. the Broadest, Most Unreasonable Interpretation

M.P.E.P. § 2131.01 provides that "[d]uring patent examination, the claims are given the broadest reasonable interpretation consistent with the specification" (emphasis provided), citing favorably, *In re Morris*, 127 F.3d 1048, 44 U.S.P.Q.2d 1023 (Fed. Cir. 1997). The specification referred to by the M.P.E.P. is the specification of the pending patent application being examined by the Examiner and not the applied reference, such as Wall et al.

For example, Claim 1 recites the feature of

the user Web service communicating with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope

The Examiner has alleged that the recited limitation of Claim 1 can be found in Wall et al. at Col. 6, lines 1-33, and Claim 1 of Wall et al. There is nothing whatsoever in the cited portions of Wall et al. that can be reasonably interpreted to disclose the feature, which portions Appellant reproduces here in full:

objects to be used to organize other objects. Objects can have attributes that are of numeric (e.g., floating point, integer and imaginary), string, boolean, vector, matrix, table and file type. Numeric attributes can be deterministic or probabilistic. Interdependencies or relationships within a model can be defined using function objects. Interfaces to existing information or systems can be defined using a combination of data objects and function objects.

In the present invention, objects can have constraints that are used to define the behavior of an object. Constraints are attributes of each object, are associated with methods of the object and corresponding constraints are checked when method invocations are made on the object. If a check indicates that a corresponding constraint is violated the method invocation is not completed, which can result in an error message/code being returned to the invoker of the method. Constraints can be placed on an object such that it will be unavailable as an input and/or an output of a model. Types of constraints include but are not limited to dependency constraints,

permissions/access control constraints, data type constraints, units constraints and message propagation constraints.

A dependency constraint can be added to a data object when the data object's value is set by a function object. While the constraint is in place, only that function object corresponding to the constraint can set the value of the data object. Objects that are dependent on other objects can only be outputs (read-only) because their values are constrained by that dependency. Once an object has a dependency constraint additional dependency relationships can not [sic] be created with other function objects. Dependency constraints prevent loops from being formed in the relationships between objects.

The above cited portions of Wall et al. discuss the basics of object oriented model including the fact that an object can have an attribute. Appellant is unable to find where Wall et al. discusses "the user Web service communicating with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope," as recited by independent Claim 1. Appellant has searched for "overlap" and was unable to find it.

The main problem here is that, instead of giving Appellant's claimed features the broadest, most reasonable interpretation, the Examiner gives Wall et al. the broadest, most unreasonable interpretation. As would be appreciated by one skilled in the art, whereas Web services need not know each other to discover each other, the legacy object-oriented model (specifically CORBA) relied on by the Examiner requires knowledge of the existence of objects. The Examiner may not divine "overlap" from either nothing or from the fact that an object can have an attribute. An accessor sentence that expresses user access scope containing dimensional extents of a security space cannot be found in the portions of Wall et al. cited by the Examiner. Appellant also cannot find a content sentence that expresses content access scope containing dimensional extents of the security space.

c. The Law Requires the Identical Invention to Be Found, But Somehow a Different Invention Is Good Enough for This Examination

As specified by M.P.E.P. § 2131.01, "the identical invention must be shown in as complete detail as is contained in the . . . claim." Citing favorably *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989) (emphasis provided). Every element of the claimed invention must be literally present, arranged as in the claim. *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989), citing *Perkin-Elmer Corp. v. Computervision Corp.*, 732 F.2d 888, 895, 221 U.S.P.Q. 669, 673; *Kallman v. Kimberly-Clark Corp.*, 713 F.2d 760, 771, 772, 218 U.S.P.Q. 781, 789 (Fed. Cir. 1983), *cert. denied*, 465 U.S. 1026, 79 L. Ed. 2d 687, 104 S. Ct. 1284 (1984). Because the Examiner has failed to show that Wall et al. discloses the identical invention as claimed by Appellant, no *prima facie* case of anticipation has been established.

The claimed subject matter requires "the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space." Nothing identical was identified by the Examiner at Col. 5, lines 1-30, of Wall et al. No expressed user access scope, no accessor sentence, and no dimensional extents of a security space were identified.

The claimed subject matter also requires "the expressed content access scope being expressed in a content sentence containing dimensional extents of the security space." Nothing identical was identified by the Examiner at Col. 5, lines 31-64, of Wall et al. No expressed content access scope, no content sentence, and no dimensional extents of the security space were identified.

The claimed subject matter further requires "the user Web service communicating with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope." Nothing identical was identified by the Examiner at Col. 6, lines 1-33, and Claim 1 of Wall et al. No "overlap" was found. It seems that

the Examiner points to a generic object-oriented software model and simply conjectures that the specific claimed subject matter can be made from it. There is a conclusion that is inescapable as revealed explicitly by the eighth required limitation of Claim 1 of Wall et al.:

identifying a user of the emergent model and assigning appropriate read, write, execute and administrative permissions to the user on a per data objects and/or function objects basis, the permissions being used to limit access to a specific subset of the data objects and/or function objects

In other words, Wall et al. advocates the tying of access control to a user on "a per data objects or function objects basis." This teaches precisely opposite from what is required by the claimed invention as recited in Claim 1 in which not only an expressed user access scope be made explicit but also an expressed content scope. Neither can be found in the teachings of Wall et al.

If this is somehow unclear to the Examiner, Appellant would like to provide this parsing of the eighth required limitation of Claim 1 of Wall et al.: Claim 1 of Wall et al. requires that a user be assigned read, write, execute, and administrative permissions on a per data objects and/or function objects basis, which permissions are used to limit access to data objects and/or function objects. In other words, in the absence of data objects or function objects, there is nothing to be assigned to the user in terms of permissions. This teaches precisely opposite from the claimed subject matter. For example, the claimed subject matter requires "the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space." No objects are mentioned in this recitation as a precondition for assigning permissions as with the eighth required limitation of Claim 1 of Wall et al.

To be further helpful to the Examiner, Appellant points to the following in the pending specification to explain the issues with coupling a user with an object. First, the pending specification explains the problems connected with a conventional permission model at page 1, line 19, to page 2, line 7:

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206 682 8100

A simple security model includes three elements: a user (who must be recognized by a computer system as an authorized user of the computer system); a piece of content that is desired by the user to view or alter, among other things; and a permission to access the piece of content by the user. This model is simple in theory but complicated for a system administrator to put into practice. Consider a situation in which a million users all want access to the piece of content. The model described above would force a system administrator to create and store a million different permissions, each corresponding with the million users, to limit access to the piece of content. Thus, more computing resources must be used—not to store content, but to process and store permissions. Not only is this economically unfeasible but the "permission explosion" itself may cripple the computer system without any undertaking by crackers.

Applicant then explains the problems connected with a role-based access control (RBAC) model at the pending specification, page 2, lines 8-20, among other places:

To avoid administrative problems associated with permission explosion, an industry standard security model called role-based access control (RBAC) was developed. In role-based access control, a system administrator programmatically assigns roles to different types of users within an organization, such as a hospital's various staffs, and permissions to secured pieces of content are granted to those assigned roles. Because there are fewer roles than there are users (multiple users may have the same role, such as a nurse role), permission explosion is avoided. A user receives permission through the role he is assigned. A user can have multiple roles, hence indirectly gaining multiple permissions to access content in a computer system. A role can inherit the characteristics of other roles so that a user with an inheriting role also gets all the permissions granted to the inherited role. Although the role-based access control model eliminates administrative problems associated with permission explosion, it creates new administrative problems, namely those associated with "role explosion."

As an example of this role explosion, a number of administrators assign a single user to a single role thus negating the RBAC model, at the pending specification, page 4, lines 3-13:

Suppose that the hospital 120 procures eye laser equipment 122. Only authorized users with the appropriate training may access and use the eye laser equipment 122 (which has a computer system to which a user must log on to operate the eye laser equipment 122). One such user is the user

102 (Dr. Murphy, the surgeon). However, the role-based access model prohibits permitting of a specific user to have access to a specific piece of content (so as to avoid permission explosion). Instead of trying to understand existing roles in the system 100, a system administrator of the hospital 120 fabricates a new role of "Dr. Murphy, the surgeon" (role 110) so as to allow only the user 102 (Dr. Murphy, the surgeon) to use the eye laser equipment. The problem, however, is that role 110 is not a role at all. If there were a thousand doctors, each doctor would require his own role, thereby creating permission explosion again.

As another example, a number of administrators do the following, at the pending specification, page 4, lines 3-13:

Now suppose the hospital 120 were to procure new CT scan equipment 124. The CT scan equipment 124 has a computer to which a user must log on in order to operate the CT scan equipment 124. Only authorized users of the hospital 120 are allowed to access and use the CT scan equipment 124. Instead of trying to understand the existing roles at the hospital 120, the system administrator creates the role of a CT scan machine (role 112), which is associated with permission 118, allowing access to user 104, the nurse, and 106, the CT scan technician. The fabricated role of the CT scan machine (role 112), however, makes no sense at all within the role-based access control model because it does not identify a group of users within an organization, but instead, identifies a machine for the convenience of the system administrator. This non-representative role 102 attenuates the security strength of the system 100 over time because system administrators have increasingly difficult time understanding various role relationships which causes difficulty in managing authorized users and eliminating unauthorized access.

Various subject matter embodiments eschew implicit roles, at the pending specification, page 12, lines 21-26:

Various embodiments of the present invention avoid role explosion by abstracting users and roles associated with users into accessors. Permissions to a piece of content are granted to accessors. Thus, if a user or a role has an accessor that is permitted to access the piece of content, the user or the role may access the piece of content. The discovery of roles eliminates permission explosion because there are fewer roles than users. The discovery of accessors eliminates role explosion because there are fewer accessors than roles. Moreover, role explosion is controlled because

permission to the piece of content is explicitly revealed through expressions rather than implicitly hidden in roles, which can be imprecise.

Various subject matter embodiments eschew explicit roles, at the pending specification, page 14, lines 7-16:

An explicit role 306 inherits the characteristics of the role class 304. The main attribute of the explicit role 306 is simply a name whose data structure is composed of a sequence of characters usually representing human-readable text. As previously discussed, explicit roles are difficult to maintain because their number may explode to an unmanageable size. An expressed role class 308 also inherits from the role class 304. The expressed role class 308 has at least two attributes: an expression type, which is preferably a set (a collection of distinct mathematical elements), and an expression, which textually describes the scope of access for a role. Various embodiments of the present invention use the expressed role class 308 to explicitly define an access scope that may encompass dozens to thousands of implied and explicit roles of a role-based access control model.

Various subject matter embodiments eschew the problems connected with the role-based access system, at the pending specification, page 17, line 22, to page 18, line 5:

A user's access scope, unlike in a role-based access control model, can be expressed independently of the access scope of the piece of content 326. The determination of when a user has permission to access a piece of content is made at access time by determining whether there is an overlap between the access scope of a user and the access scope of a piece of content. This decoupling is possible in various embodiments of the present invention allowing the piece of content 326 to be granted to classes of users, such as users 318B-318C, without ever needing to form an explicit relationship tying users 318B-318C to the piece of content 326. Access scopes of users may be completely defined via expressions without needing to determine which pieces of content are accessible via certain access scopes, and, at the same time, access scopes of pieces of content may be completely defined via expressions without any reference to classes of users, such as users 318B-318C. This reduces the size of the permission space, hence reducing exponential role explosion and simplifying administration of the system 316.

To repeat for emphasis, Claim 1, the Abstract, and even the Summary of the Invention of Wall et al. all require "assigning appropriate read, write, execute and administrative permissions to the user on a per data objects and/or function objects basis, the permissions being used to limit access to a specific subset of the data objects and/or function objects." The claimed subject matter of Claim 1 does not assign permissions to a user on "a per data objects and/or function objects basis." The reason, as would be appreciated by one with ordinary skill in the art, is that there may be no knowledge of a data object or a function object in the paradigm of Web services. In contrast, the permissions that are assigned to a user in the legacy system of Wall et al. occur on "a per data objects and/or function objects basis." If there is no knowledge of a data object or a function object, there can be no assignment of permissions to a particular user in the system of Wall et al.

2. Dependent Claims 4, 9, and 16

Claim 4 is dependent on Claim 1 and recites that a compiler Web service is configured for compiling the accessor sentence and the content sentence into binary sentences, each binary sentence comprising binary phrases. Claim 9 is dependent on Claim 6 and recites compiling the user access scope sentence and the content access scope sentence by a sentence compiler Web service into binary sentences, each binary sentence having binary phrases, each binary phrase being a compiled dimensional extent. Claim 16 is dependent on Claim 13 and recites compiling the user access scope sentence and the content access scope sentence by a sentence compiler Web service into binary sentences, each binary sentence having binary phrases, each binary phrase being a compiled dimensional extent. The Examiner has argued that Wall et al. teaches these features of the claimed subject matter at Col. 7, lines 40-52. This cannot be correct as that portion is presented here in full:

The object reference table can contain references to data objects and function objects. The expression text can contain portions that specify

method calls on the objects referenced in the object reference table. The solver evaluates the expression text, which may result in changes to objects referenced by the function object. Solvers can be implemented using compilers, interpreters or entire legacy applications. The function object solves the expression when a change message is received from one of the objects referenced by the function object. Function objects can also be solved manually by a user. Dependency constraints are optionally placed on data objects that have their values set by the expression of a function object.

Appellant incorporates by reference the arguments discussed in connection with Independent Claim 1, as if the discussed arguments were set forth here in full.

3. Dependent Claim 5

Claim 5 is dependent on Claim 4 and recites an evaluator Web service for comparing the accessor sentence and the content sentence to determine whether to grant access to the user Web service so that the user Web service can access the piece of content. The Examiner has argued that Wall et al. teaches this feature of the claimed subject matter at Col. 7, lines 15-64. This cannot be correct as that portion is presented here in full:

A unit is a particular physical quantity, defined and adopted by convention, with which other particular quantities of the same kind are compared to express their value. The International System of Units (SI) defines seven base units:

TABLE-US-00001 length meter, mass kilogram, time second, electric current ampere, thermodynamic temperature, amount of substance mole, luminous intensity candela.

Function objects provide behavior by providing expressions that can relate data objects and function objects thereby creating networks of linked function objects and data objects. Function objects have a name, an expression, an object reference table and a solver attribute as well as access policy and trigger message propagation constraints. The expression can be thought of as a function with a plurality of inputs and output objects, which is evaluated by the solver. The expression text can be defined using various programming languages including Basic, C, C++ and Java, among others. The object reference table can contain references to data objects and function objects. The expression text can contain

portions that specify method calls on the objects referenced in the object reference table. The solver evaluates the expression text, which may result in changes to objects referenced by the function object. Solvers can be implemented using compilers, interpreters or entire legacy applications. The function object solves the expression when a change message is received from one of the objects referenced by the function object. Function objects can also be solved manually by a user. Dependency constraints are optionally placed on data objects that have their values set by the expression of a function object.

An equivalence function object is an example of a function object implemented with the present invention. The following simplified example illustrates an equivalence function object that keeps several data objects equivalent (i.e., if one of the data object values changes, then the other data object values are changed such that their values match). In this example, the equivalence function is used in combination with a number data object, further providing an example of a message propagation constraint being applied.

The above portions of Wall et al. discuss "unit" as a physical quantity and function objects. No evaluator Web service can be found.

Appellant incorporates by reference the arguments discussed in connection with Independent Claim 1, and dependent Claims 4, 9, 16 as if the discussed arguments were set forth here in full.

4. Independent Claim 6

Claim 6 succinctly defines a computer-implemented method for evaluating the scope of a content access request by a user. The method comprises requesting a discovery framework by a user Web service to access a piece of content represented by a content Web service. The method further comprises notifying the content Web service by the discovery framework of the access request by the user Web service. The method further comprises requesting the discovery framework by the content Web service for an access evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content. The access scope of the user Web service is

conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service without using predicate rules, sequencing rules, and implication rules. The access scope of the user Web service is expressed by the first expression in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service is expressed in a content access scope sentence containing dimensional extents of the security space. Because the Examiner has failed to show that Wall et al. discloses every element of the claimed subject matter, no *prima facie* case of anticipation has been established.

Each claim limitation is neither taught nor suggested by Wall et al. The Examiner cited so many portions of Wall et al., such as Col. 5, lines 1-30; Col. 6, lines 1-33; Col. 5, lines 31-64; and even Claim 1 of Wall et al., but none of them teaches or suggests the claimed subject. As an example, the system of Wall et al. completely lacks the feature of:

the access scope of the user Web service being conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service

as recited in independent Claim 6. No where does Wall et al. disclose this feature. Instead Wall et al. teaches in the opposite direction, which Appellant recites here in full (and which is found at the Abstract, Summary of The Invention, and Claim 1 of Wall et al.):

identifying a user of the emergent model and assigning appropriate read, write, execute and administrative permissions to the user on a per data objects and/or function objects basis, the permissions being used to limit access to a specific subset of the data objects and/or function objects

In other words, the permissions of a user of Wall et al. are dependent on data objects or function objects. They are not independent of one another. These teachings of Wall et al. cannot be a basis relied on by Examiner to support the claimed feature "the access scope of the user Web service being conveyed in a first expression independently from a second expression that conveys

the access scope of the content Web service," as recited by Claim 6 among many other claimed features.

Appellant incorporates by reference the arguments discussed in connection with Independent Claim 1, and dependent Claims 4, 5, 9, 16 as if the discussed arguments were set forth here in full.

5. Dependent Claims 10 and 17

Claim 10 is dependent on Claim 9 and recites evaluating the binary sentences by the access evaluator service, the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence. Claim 17 is dependent on Claim 16 and recites evaluating the binary sentences by the access evaluator service, the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence. The Examiner has argued that Wall et al. teaches these features of the claimed subject matter at Col. 7, lines 40-64. This cannot be correct as that portion is presented here in full:

The expression text can contain portions that specify method calls on the objects referenced in the object reference table. The solver evaluates the expression text, which may result in changes to objects referenced by the function object. Solvers can be implemented using compilers, interpreters or entire legacy applications. The function object solves the expression when a change message is received from one of the objects referenced by the function object. Function objects can also be solved manually by a user. Dependency constraints are optionally placed on data objects that have their values set by the expression of a function object.

An equivalence function object is an example of a function object implemented with the present invention. The following simplified example illustrates an equivalence function object that keeps several data objects equivalent (i.e., if one of the data object values changes, then the other data object values are changed such that their values match). In this example, the equivalence function is used in combination with a number

data object, further providing an example of a message propagation constraint being applied.

Appellant is unable to find where the recited feature "the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence" can be found in Wall et al., especially in the above cited portions by the Examiner. No binary phrase, no comparison on each binary phrase of one sentence to another corresponding sentence, and no formation of a resultant binary sentence can be found.

Appellant incorporates by reference the arguments discussed in connection with Independent Claims 1, 6, and dependent Claims 4, 5, 9, 16 as if the discussed arguments were set forth here in full.

6. Dependent Claims 11 and 18

Claim 11 is dependent on Claim 10 and recites granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero. Claim 18 is dependent on Claim 17 and recites granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero. The Examiner has argued that Wall et al. teaches these features of the claimed subject matter at Col. 6, lines 1-33, of Wall et al. There is nothing whatsoever in the cited portions of Wall et al. that can be reasonably interpreted to disclose the features, which portions Appellant reproduces here in full:

objects to be used to organize other objects. Objects can have attributes that are of numeric (e.g., floating point, integer and imaginary), string, boolean, vector, matrix, table and file type. Numeric attributes can be deterministic or probabilistic. Interdependencies or relationships within a model can be defined using function objects. Interfaces to existing information or systems can be defined using a combination of data objects and function objects.

In the present invention, objects can have constraints that are used to define the behavior of an object. Constraints are attributes of each object,

are associated with methods of the object and corresponding constraints are checked when method invocations are made on the object. If a check indicates that a corresponding constraint is violated the method invocation is not completed, which can result in an error message/code being returned to the invoker of the method. Constraints can be placed on an object such that it will be unavailable as an input and/or an output of a model. Types of constraints include but are not limited to dependency constraints, permissions/access control constraints, data type constraints, units constraints and message propagation constraints.

A dependency constraint can be added to a data object when the data object's value is set by a function object. While the constraint is in place, only that function object corresponding to the constraint can set the value of the data object. Objects that are dependent on other objects can only be outputs (read-only) because their values are constrained by that dependency. Once an object has a dependency constraint additional dependency relationships can not [sic] be created with other function objects. Dependency constraints prevent loops from being formed in the relationships between objects.

The above cited portions of Wall et al. discuss the basics of object oriented model including the fact that an object can have an attribute. Appellant is unable to find where Wall et al. discusses "granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero," as recited by dependent Claim 11. Appellant has searched for "zero" and was unable to find it in the recited portions of Wall et al.

Appellant incorporates by reference the arguments discussed in connection with Independent Claims 1, 6, and dependent Claims 4, 5, 9, 10, 16, 17, and 18 as if the discussed arguments were set forth here in full.

7. Dependent Claims 12 and 19

Claim 12 is dependent on Claim 11 and recites denying access to the user Web service if one binary phrase of the resultant binary sentence is equal to zero. Claim 19 is dependent on Claim 18 and recites denying access to the user Web service if one binary phrase of the resultant binary sentence is equal to zero. The Examiner has argued that Wall et al. teaches these features

of the claimed subject matter at Col. 6, lines 1-33, of Wall et al. There is nothing whatsoever in the cited portions of Wall et al. that can be reasonably interpreted to disclose the features, which portions Appellant reproduces here in full:

objects to be used to organize other objects. Objects can have attributes that are of numeric (e.g., floating point, integer and imaginary), string, boolean, vector, matrix, table and file type. Numeric attributes can be deterministic or probabilistic. Interdependencies or relationships within a model can be defined using function objects. Interfaces to existing information or systems can be defined using a combination of data objects and function objects.

In the present invention, objects can have constraints that are used to define the behavior of an object. Constraints are attributes of each object, are associated with methods of the object and corresponding constraints are checked when method invocations are made on the object. If a check indicates that a corresponding constraint is violated the method invocation is not completed, which can result in an error message/code being returned to the invoker of the method. Constraints can be placed on an object such that it will be unavailable as an input and/or an output of a model. Types of constraints include but are not limited to dependency constraints, permissions/access control constraints, data type constraints, units constraints and message propagation constraints.

A dependency constraint can be added to a data object when the data object's value is set by a function object. While the constraint is in place, only that function object corresponding to the constraint can set the value of the data object. Objects that are dependent on other objects can only be outputs (read-only) because their values are constrained by that dependency. Once an object has a dependency constraint additional dependency relationships can not [sic] be created with other function objects. Dependency constraints prevent loops from being formed in the relationships between objects.

The above cited portions of Wall et al. discuss the basics of object oriented model including the fact that an object can have an attribute. Appellant has searched for "zero" and was unable to find that word in the recited portions of Wall et al.

Appellant incorporates by reference the arguments discussed in connection with Independent Claims 1, 6, and dependent Claims 4, 5, 9, 10, 11, 16, 17, and 18 as if the discussed arguments were set forth here in full.

8. Independent Claim 13

Claim 13 succinctly defines a tangible and storable computer-readable medium having computer-readable instructions that implement a method for evaluating the scope of a content access request by a user. The method comprises requesting a discovery framework by a user Web service to access a piece of content represented by a content Web service. The access scope of the user Web service is expressed in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service being expressed in a content access scope sentence containing dimensional extents of the security space. The method comprises notifying the content Web service by the discovery framework of the access request by the user Web service. The method further comprises requesting the discovery framework by the content Web service for an access evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content without forming an explicit relationship tying the user Web service to the content Web service via predicate rules, sequence rules, and implication rules. Because the Examiner has failed to show that Wall et al. discloses every element of the claimed subject matter, no prima facie case of anticipation has been established.

Each claim limitation is neither taught nor suggested by Wall et al. The Examiner cited so many portions of Wall et al., such as Col. 5, lines 1-30; Col. 6, lines 1-33; Col. 5, lines 31-64; and even Claim 1 of Wall et al., but none of them teaches or suggests the claimed subject. As one example, the system of Wall et al. completely lacks the feature of:

evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content without forming an explicit relationship tying the user Web service to the content Web service

as recited in independent Claim 13. Nowhere does Wall et al. disclose this feature. Instead Wall et al. teaches in the opposite direction, which Appellant recites here in full (and which is found at the Abstract, Summary of The Invention, and Claim 1 of Wall et al.):

identifying a user of the emergent model and assigning appropriate read, write, execute and administrative permissions to the user on a per data objects and/or function objects basis, the permissions being used to limit access to a specific subset of the data objects and/or function objects

In other words, the permissions of a user of Wall et al. are explicitly tied to data objects or function objects. They are not independent of one another. These teachings of Wall et al. cannot be a basis relied on by Examiner to support the claimed feature "evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content without forming an explicit relationship tying the user Web service to the content Web service," as recited by Claim 13 among many other claimed features.

Appellant incorporates by reference the arguments discussed in connection with Independent Claims 1, 6, and dependent Claims 4, 5, 9, 10, 11, 12, 16, 17, 18, and 19 as if the discussed arguments were set forth here in full.

C. A Recap of the Claimed Invention Clearly Shows That Wall et al., Does Not Teach, Let Alone Render Unpatentable, the Claimed Invention

Clearly Wall et al. fails to teach or suggest the subject matter of Claim 1. More specifically, this reference fails to teach or suggest "the user Web service communicating with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope," as recited in Claim 1, among other claim limitations. As will be appreciated from the foregoing discussion, the applied and cited reference

neither teaches nor suggests the subject matter of Claim 1. As a result, Appellant submits that Claim 1 is clearly allowable in view of the teachings of the reference.

With respect to dependent Claims 4 and 5, all of which depend directly or indirectly from Claim 1, it is also clear that the subject matter of these claims is neither taught nor suggested by the applied and cited reference, namely, Wall et al., particularly when the features are considered in combination with the recitations of the claims from which these claims individually depend. In summary, Claims 4 and 5 are submitted to be allowable for reasons in addition to the reasons why Claim 1 is submitted to be allowable.

Independent Claim 6 is directed to a method. The applied and cited reference fails to teach or suggest "the access scope of the user Web service being conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service," as recited in Claim 6, among other features. For generally the same reasons discussed above with respect to Claim 1, Appellant submits that the subject matter of Claim 6 is neither taught nor suggested by the applied and cited reference.

With respect to dependent Claims 9-12, all of which depend directly or indirectly from Claim 6, it is also clear that the subject matter of these claims is neither taught nor suggested by the applied and cited references. Claims 9-12 all have limitations that are clearly neither taught nor suggested by the applied and cited reference, particularly when the features are considered in combination with the recitations of the claims from which these claims individually depend. In summary, Claims 9-12 are submitted to be allowable for reasons in addition to the reasons why Claim 6 is submitted to be allowable.

Independent Claim 13 is directed to a computer-readable media containing computer-executable instructions that, when executed, carry out the method for evaluating the scope of a content access request by a user. The applied and cited reference fails to teach "an access

evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content without forming an explicit relationship tying the user Web service to the content Web service," as recited in Claim 13, among other features. For generally the same reasons discussed above with respect to Claims 1 and 6, Appellant submits that the subject matter of Claim 13 is neither taught nor suggested by the applied and cited reference, and thus, Claim 13 is also allowable.

In light of the foregoing remarks, it is clear that none of the applied and cited references teaches, let alone renders unpatentable, the claimed inventions recited in Claims 1-25. The applied and cited references are directed to decoding keys that are different from the claimed invention, use a sequential detection technique that is neither can accommodate a single key result or simultaneous actuation of key combinations, employ keys that are not side by side, work in an entirely different manner from the present invention, or simply have nothing to do with the claimed invention. The claimed invention is directed to an entirely different concept and solution.

With respect to dependent Claims 16-19, all of which depend directly or indirectly from Claim 13, it is also clear that the subject matter of these claims is neither taught nor suggested by the applied and cited references. Claims 16-19 all have limitations that are clearly neither taught nor suggested by the applied and cited reference, particularly when the features are considered in combination with the recitations of the claims from which these claims individually depend. In summary, Claims 16-19 are submitted to be allowable for reasons in addition to the reasons why Claim 13 is submitted to be allowable.

In view of the foregoing remarks, Appellants submit that all of the claims in the present application are patentably distinguishable over the teachings of Wall et al. Therefore, it is

submitted that the rejections of Claims 1, 4-6, 9-13, and 16-19 were erroneous, and reversal of the rejections is respectfully requested.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

VIII. CLAIMS APPENDIX

1. A networked system for accessing a piece of content, comprising:

a user Web service for representing a user having an expressed user access scope, the expressed user access scope being expressed in an accessor sentence containing dimensional extents of a security space; and

a content Web service for representing a piece of content having an expressed content access scope, the expressed content access scope being expressed in a content sentence containing dimensional extents of the security space, the user Web service communicating with the content Web service to access the piece of content when the expressed user access scope overlaps with the expressed content access scope without using predicate rules, sequencing rules, and implication rules.

2-3. (Canceled)

4. The networked system of Claim 1, further comprising a compiler Web service for compiling the accessor sentence and the content sentence into binary sentences, each binary sentence comprising binary phrases.

5. The networked system of Claim 4, further comprising an evaluator Web service for comparing the accessor sentence and the content sentence to determine whether to grant access to the user Web service so that the user Web service can access the piece of content.

6. A computer-implemented method for evaluating the scope of a content access request by a user, the method comprising:

requesting a discovery framework by a user Web service to access a piece of content represented by a content Web service;

notifying the content Web service by the discovery framework of the access request by the user Web service; and

requesting the discovery framework by the content Web service for an access evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content, the access scope of the user Web service being conveyed in a first expression independently from a second expression that conveys the access scope of the content Web service without using predicate rules, sequencing rules, and implication rules, the access scope of the user Web service being expressed by the first expression in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service being expressed in a content access scope sentence containing dimensional extents of the security space.

7-8. (Canceled)

9. The method of Claim 6, further comprising compiling the user access scope sentence and the content access scope sentence by a sentence compiler Web service into binary sentences, each binary sentence having binary phrases, each binary phrase being a compiled dimensional extent.

10. The method of Claim 9, further comprising evaluating the binary sentences by the access evaluator service, the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence.

11. The method of Claim 10, further comprising granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero.

12. The method of Claim 11, further comprising denying access to the user Web service if one binary phrase of the resultant binary sentence is equal to zero.

13. A tangible and storable computer-readable medium having computer-readable instructions that implement a method for evaluating the scope of a content access request by a user, the method comprising:

requesting a discovery framework by a user Web service to access a piece of content represented by a content Web service, the access scope of the user Web service being expressed in a user access scope sentence containing dimensional extents of a security space and the access scope of the content Web service being expressed in a content access scope sentence containing dimensional extents of the security space;

notifying the content Web service by the discovery framework of the access request by the user Web service; and

requesting the discovery framework by the content Web service for an access evaluator Web service to evaluate whether an access scope of the user Web service overlaps with an access scope of the content Web service to grant access to the piece of content without forming an explicit relationship tying the user Web service to the content Web service via predicate rules, sequence rules, and implication rules.

14-15. (Canceled)

16. The method of Claim 13, further comprising compiling the user access scope sentence and the content access scope sentence by a sentence compiler Web service into binary sentences, each binary sentence having binary phrases, each binary phrase being a compiled dimensional extent.

17. The method of Claim 16, further comprising evaluating the binary sentences by the access evaluator service, the act of evaluating including comparing each binary phrase of a first binary sentence with each corresponding binary phrase of a second binary sentence to form a resultant binary sentence.

18. The method of Claim 17, further comprising granting access to the user Web service if each binary phrase of the resultant binary sentence is greater than zero.

19. The method of Claim 18, further comprising denying access to the user Web service if one binary phrase of the resultant binary sentence is equal to zero.

IX. EVIDENCE APPENDIX

None.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682 8100

X. RELATED PROCEEDINGS APPENDIX

None.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS^{PLLC}

D.C. Peter Chu
Registration No. 41,676
Direct Dial No. 206.695.1636

DPC:jljg

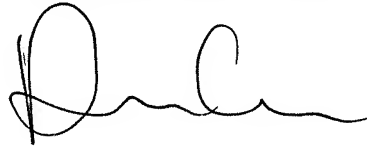
LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

X. RELATED PROCEEDINGS APPENDIX

None.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS^{PLLC}

A handwritten signature in black ink, appearing to read 'D.C. Peter Chu', with a stylized, cursive script.

D.C. Peter Chu
Registration No. 41,676
Direct Dial No. 206.695.1636

DPC;jlg

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^{PLLC}
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206 682 8100